Midterm Exam ECE 448
Spring 2020
Wednesday, March 4
15 points

Instructions:
Zip all your deliverables into an archive <last_name>.zip and submit it through Blackboard no later than Wednesday, March 4, 10:05 PM EST.

The EXAM circuit calculates and outputs four different sequences of 8-bit numbers specified below:

1. Padovan Sequence

The Padovan sequence is the sequence of integers defined by the initial values

$$P_0 = P_1 = P_2 = 1,$$

and the recurrence relation

$$P_n = P_{n-2} + P_{n-3}.$$

The 8-bit values of this sequence are

1, 1, 1, 2, 2, 3, 4, 5, 7, 9, 12, 16, 21, 28, 37, 49, 65, 86, 114, 151, 200.

Values of this sequence, starting from P_3, can be calculated using the following pseudocode:

```
P_nm3 = 1
P_nm2 = 1
P_nm1 = 1
n=3
while (Pnm2 + Pnm3 < 256) do
   Pn = Pnm2 + Pnm3
   n = n+1
   P_nm3 = P_nm2
   P_nm2 = P_nm1
   P_nm1 = P_n
end while
```

Notation:

`P_n` is the n-th element of the Padovan sequence
`P_nm3` is the element at the position n-3
`P_nm2` is the element at the position n-2
`P_nm1` is the element at the position n-1

2. Sequence based on a bit counter

The first element of the sequence is set to 1.
The next element of the sequence is equal to the sum of the value of the current element and the number of ones in this element.

$$F_{n+1} = F_n + F_n(7) + F_n(6) + \cdots + F_n(0)$$

Notation:

F_n is the current element of the sequence
$F_n(i)$ is the i-th bit of F_n
F_{n+1} is the next element of the sequence

The sequence should be repeated until

$$F_{n+1} = F_n \quad or \quad F_{n+1} > 2^8\text{-}1$$

The elements of this sequence include:

1, 2, 3, 5, 7, 10, 12, 14, 17, 19, 22, 25, 28, 31, ….

3. Sequence based on a 3MSB-3LSB complement

The first element of the sequence is set to 1.
The next element of the sequence is equal to the sum of the value of

- the current element and
- the one's complement of its 3 most significant bits if the middle two bits are the same or the one's complement of its 3 least significant bits if the middle two bits are different.

$$F_{n+1} = F_n + \left[F_n(7..5) \cdot \left(F_n(3) \oplus F_n(4)\right)'\right]' + \left[F_n(2..0) \cdot \left(F_n(3) \oplus F_n(4)\right)\right]' + 8$$

Alternatively, we can define F_{n+1} as follows:

$F_{n+1} = F_n + [F_n(7..5)]' + 8$ if $F_n(3) \oplus F_n(4) = 0$
$F_{n+1} = F_n + [F_n(2..0)]' + 8$ if $F_n(3) \oplus F_n(4) = 1$

Notation:

Y' is a one's complement of Y
$F_n(7..5)$ are the three most significant bits of F_n treated as a 3-bit unsigned integer
$F_n(2..0)$ are the three least significant bits of F_n treated as a 3-bit unsigned integer
$F_n(3)$ and $F_n(4)$ are two middle bits of F_n

The sequence should be repeated until

$$F_{n+1} = F_n \ \ or \ \ F_{n+1} > 2^8\text{-}1$$

The elements of this sequence include:

1, 16, 31, 46, 55, ...

4. Sequence based on an addition of a constant modulo a prime

The first element of the sequence is set to 1.
The next element of the sequence is given by

$$F_n = (F_{n-1} + 46)\ mod\ 251$$

Notation:

A mod N represents a remainder from the division of A by N, e.g., 33 mod 5 = 3, 257 mod 251 = 6.

The sequence should be repeated until

$$F_{n+1} = 1$$

The elements of this sequence include:

1, 47, 93, 139, 185, 231, 26, 72, 118, 164, 210, 5, 51, …

After the start input is active for one clock period, the circuit should calculate and output all four sequences, one after another, then activate the output ready and wait for another start input.

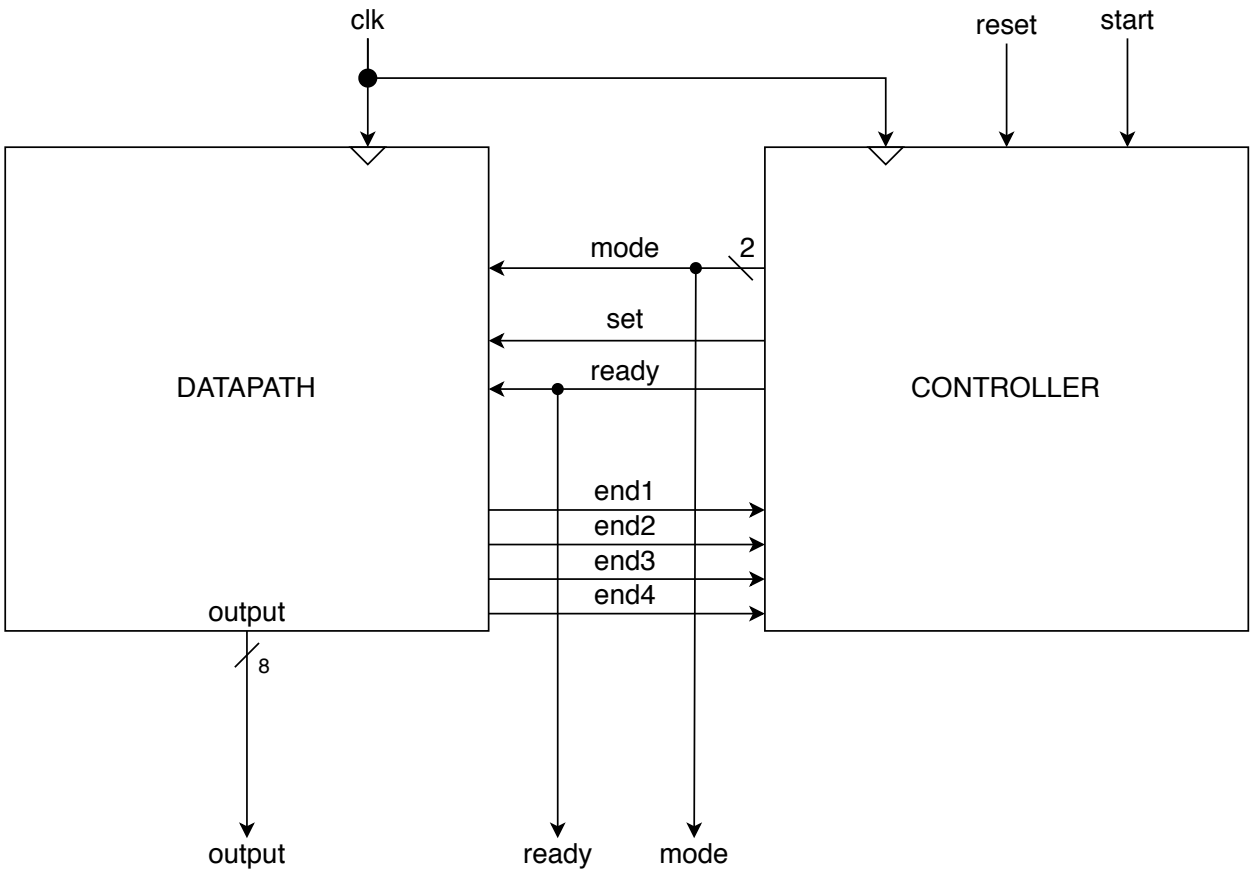
The circuit is specified below using its

- Table of input/output ports
- Interface with the division into the Datapath and Controller
- Block diagram of the Datapath
- ASM chart of the Controller.

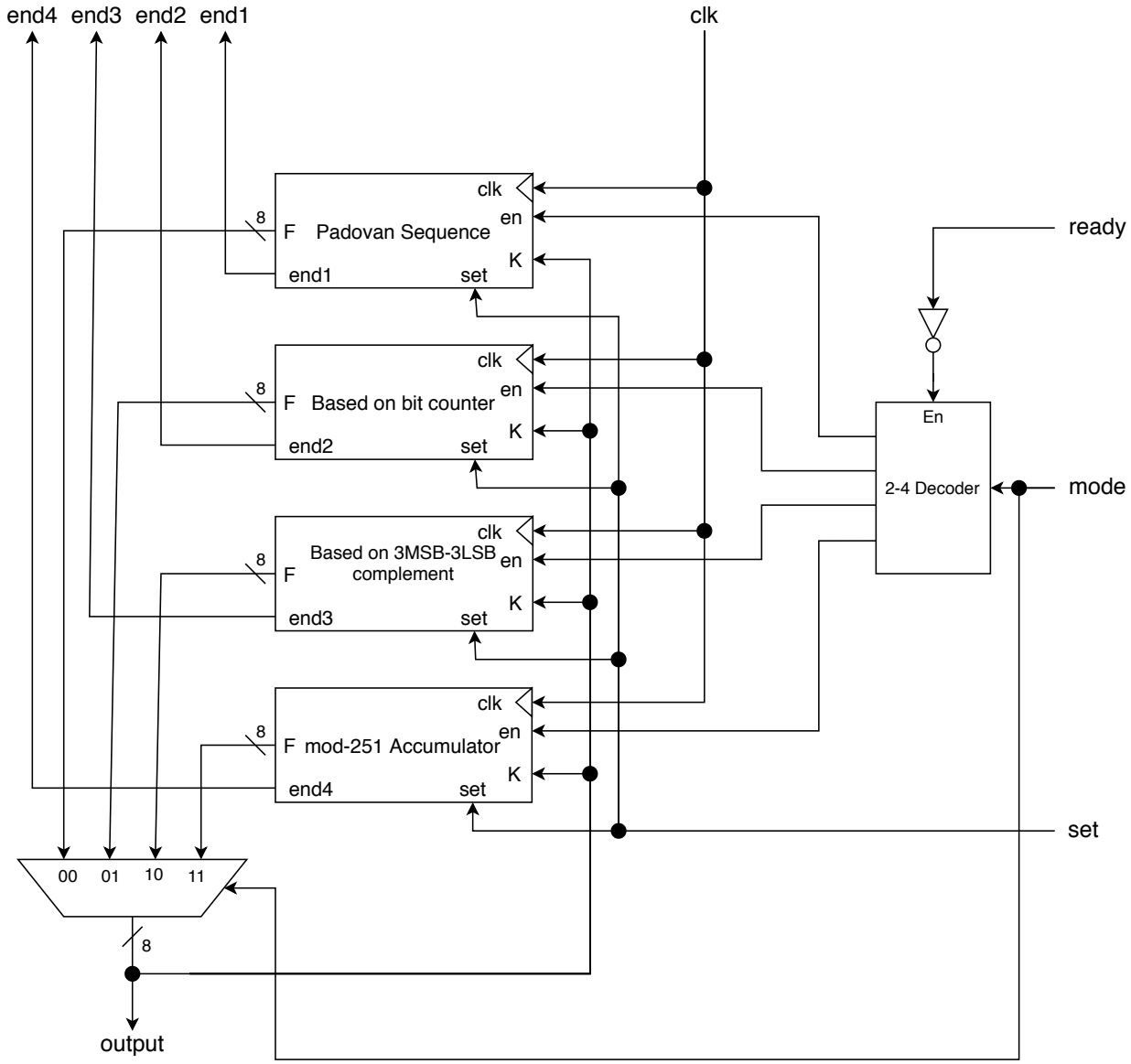
Table of input/output ports

Port	Width	Meaning
clk	1	System clk
reset	1	System reset
start	1	Input used to start the generation of four sequences
ready	1	Output indicating that the circuit is ready for a start signal
output	8	Value of the current element of the sequence
mode	2	The code of the current sequence being executed: 00-Padovan Sequence, 01-Sequence based on a bit counter 10-Sequence based on a 3MSB-3LSB complement 11-Sequence based on an addition of a constant modulo a prime

Top-level circuit interface

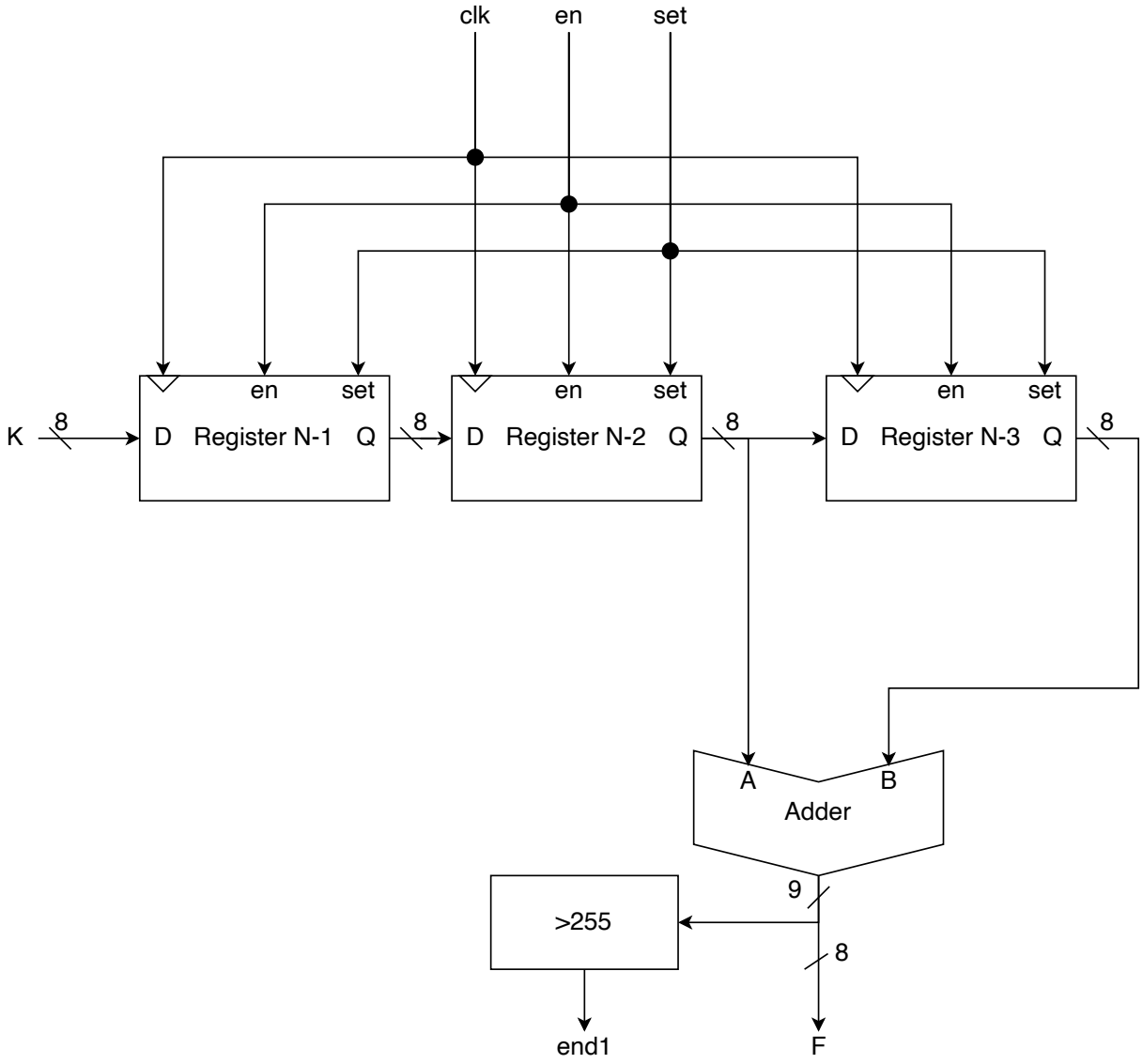


Block diagram of the Datapath

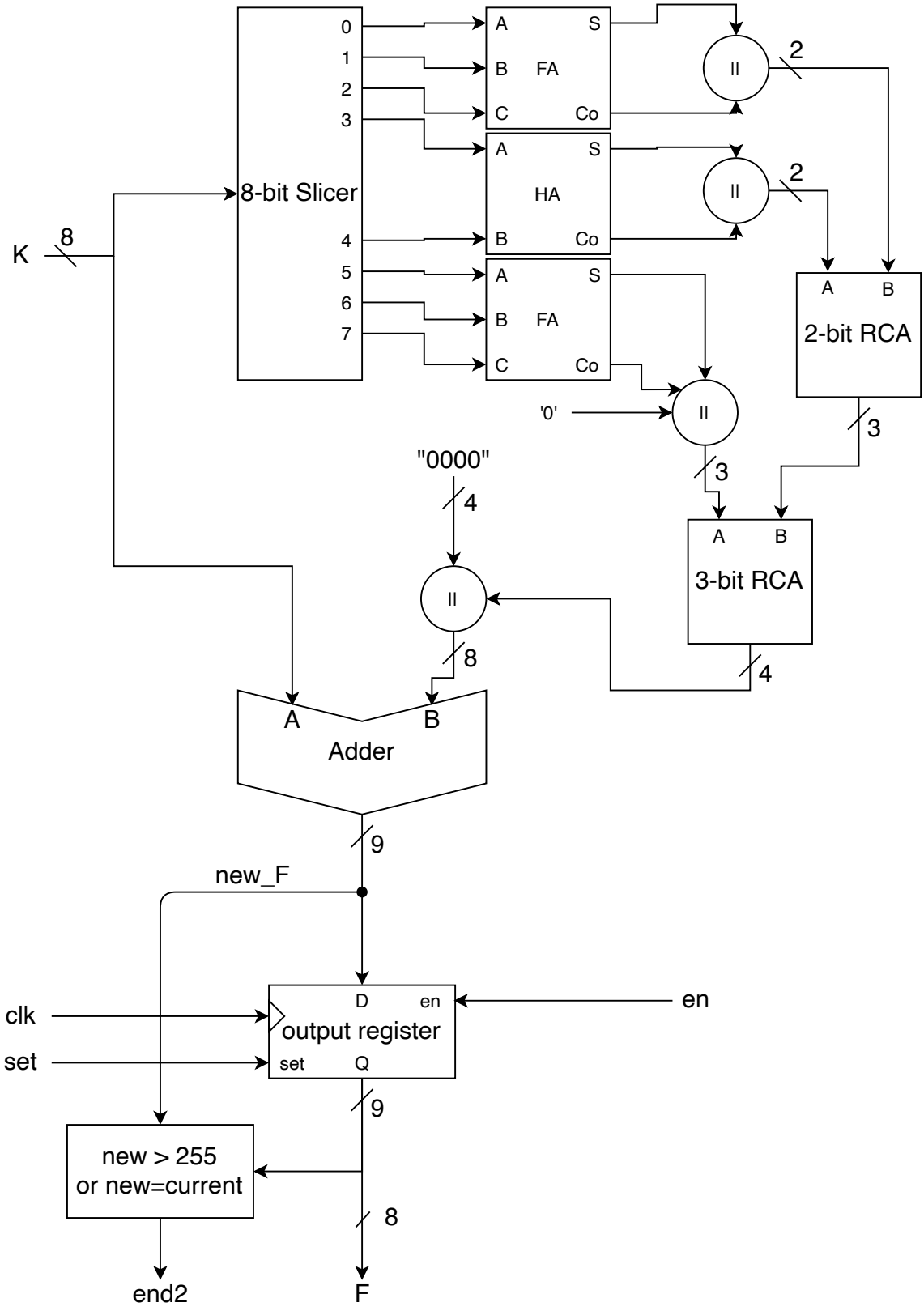


The set input initializes all internal 8-bit registers to “00000001”.

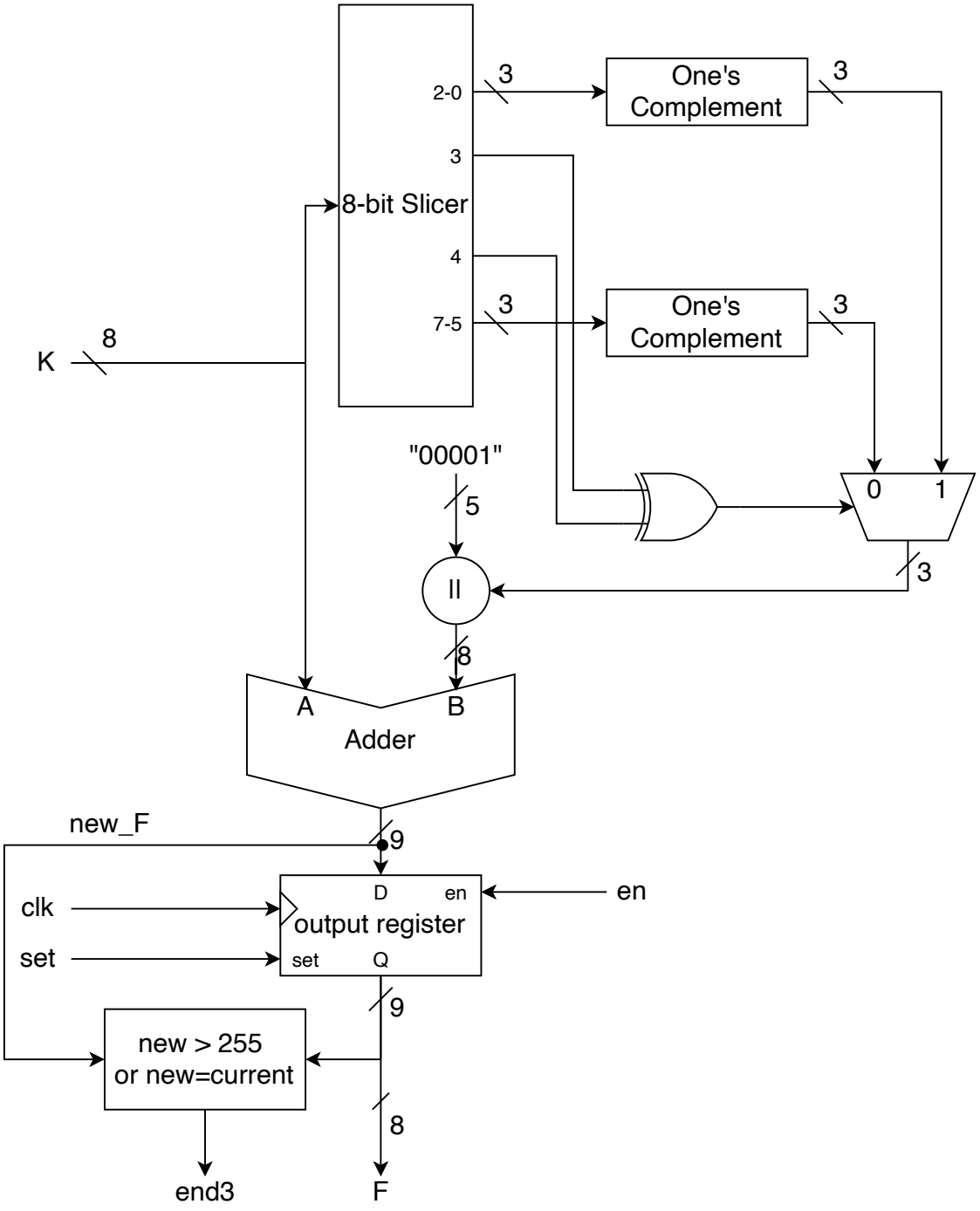
Block diagram of the generator of Padovan Sequence



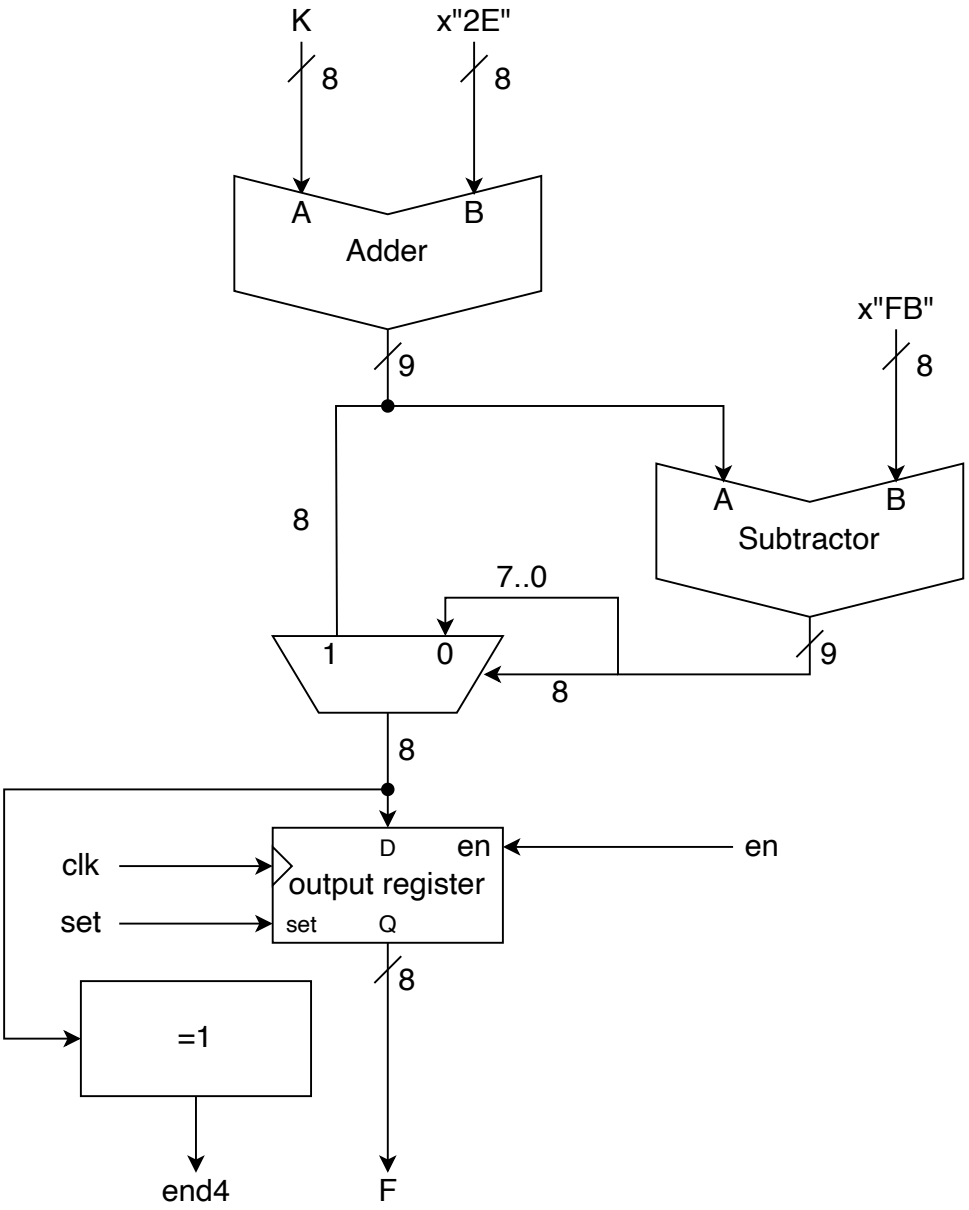
Block diagram of the generator of a sequence based on a bit counter



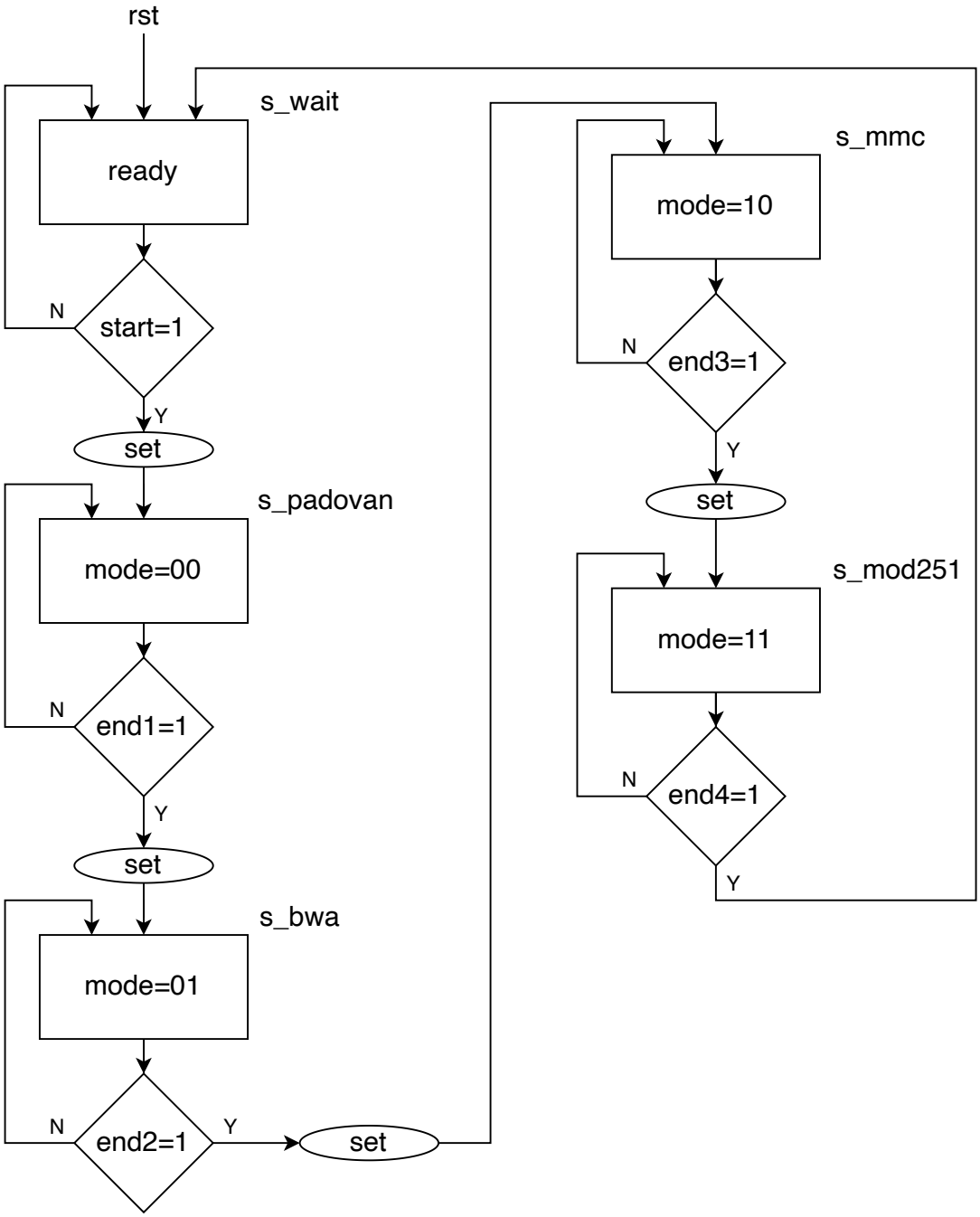
Block diagram of the generator of a sequence based on a 3MSB-3LSB complement



Block diagram of the generator of a sequence based on an addition of a constant modulo a prime



ASM Chart


rst
s_wait
ready
start=1
N
Y
set
s_padovan
mode=00
end1=1
N
Y
set
s_bwa
mode=01
end2=1
N
Y
set
s_mmc
mode=10
end3=1
N
Y
set
s_mod251
mode=11
end4=1
N
Y

Tasks:

Perform the following tasks:

1. Write a synthesizable VHDL code representing the circuit.
2. Write a testbench verifying the operation of your circuit.
3. Perform functional simulation of your circuit and use it to debug your VHDL code. Take a printout of the waveform showing the entire operation.
4. Synthesize your circuit.
5. Implement your circuit using
    a. FPGA family: Artix-7
    b. Part name: XC7A35TCPG236-1
    c. Speed Grade: -1
6. Based on the implementation reports, determine the number of CLB slices, LUTs, flip-flops, and pins used by the circuit.

Deliverables:

1. VHDL code of your entire circuit.
2. VHDL code of your testbench.
3. Timing waveforms from the functional simulation demonstrating outputs generated by your circuit.
4. FPGA resource utilization (as defined in Task 6 above).